(4)

Technical Document 1936
August 1990

Mathematical Techniques for the Performance Oriented Design (POD) Tool

A. Sterrett

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152-5000

J. D. FONTANA, CAPT, USN
Commander

R. M. HILLYER
Technical Director

ADMINISTRATIVE INFORMATION

This work was performed by Code 411, Naval Ocean Systems Center (NOSC), under the Software Engineering Technology Project.

Released by
D. Hayward, Head
Computer Systems Software
and Technology Branch

Under authority of
A. G. Justice, Head
Information Processing and
Displaying Division

CONTENTS

CONTENTS (continued)

FIGURES

INTRODUCTION

This document describes the mathematical techniques used by the NOSC Performance Oriented Design (POD) tool. The POD evaluates and predicts the performance of Computer Systems in terms of Response Time, Throughput, and Utilization. An algorithm that is extensible to distributed systems is also provided.

MEAN VALUE ANALYSIS (MVA)

Mean Value Analysis (MVA) is a technique for analyzing queueing networks that represent computer systems. MVA provides a developer with important performance measures such as Response Time, Throughput, Queue Length, and Device Utilization.

MVA NOTATIONS AND DEFINITIONS

Notations

Model Inputs

- A_c – Customer arrival rate for class c (for open classes) or
- N_c – Number of class c terminals (for closed classes), i.e., number of terminals from which class c customers originate.
- Z_c – Class c think time.
- $S_{c,d}$ – Class c service demand at queueing center/device d.

Model Outputs

Class Measures

- R_c – Response Time for class c.
- X_c – Throughput for class c.

Device Measures

- $U_{c,d}$ – Class c utilization at device d.
- $Q_{c,d}$ – Class c average queue length at device d.

Definitions

The following is a list of commonly used MVA terms that are also shown in figure 1, performance measures.

Class

A class is a group of customers with each customer of that group having the same workload intensities (A_c, N_c, or N_c and Z_c) and service demand ($S_{c,d}$). A class MUST define the service demand ($S_{c,d}$) at each device used.

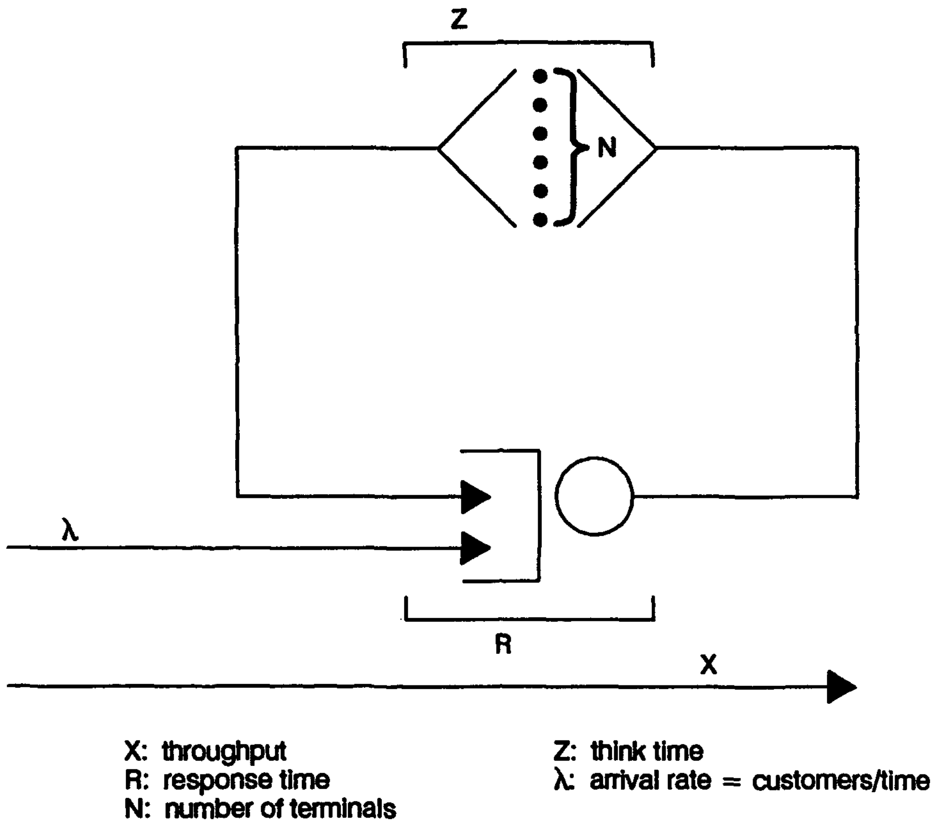


Figure 1. Performance measures.

Throughput

Throughput is the number of customers serviced by a device. x_j is device throughput.

Response Time

Response Time is the time a customer spends at a device r_i.

System Throughput

System Throughput is the number of customers serviced by the system within a given period of time. $X = \sum_i x_i$.

System Response Time

System Response Time is the sum time a customer spends at each device. $R = \sum_i r_i$.

Device Utilization

Device Utilization is a ratio between the busy time (time a device is being used) and the available time of the device. This ratio describes the extent of utilization of a device. Utilization is defined as less than equal to 1.0; therefore, 0.5 utilization means half of the device's possible capacity is being used.

Device Queue Length

Device Queue Length is the number of customers waiting for service in a particular queueing center.

Think Time

Think Time represents the time a user spends before submitting a customer.

Arrival Rate

Arrival Rate is the rate of customers arriving at a queue per unit time.

Queueing Center

Queueing center is a means of representing a resource and waiting area (queue). In this queueing center, a customer arrives for service and waits in line (the queue) until the resource is available. This may represent a bank teller line or a computer CPU. See figure 2.

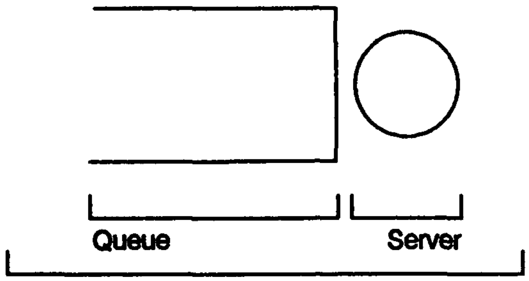


Figure 2. Queueing center.

Queueing Network

A Queueing Network is a set of queueing centers. Figure 3 is an example of a queueing network with three queueing centers.

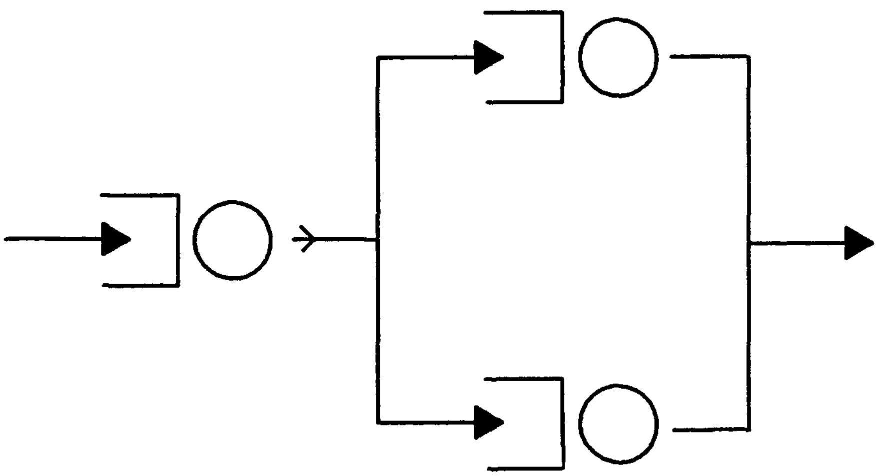


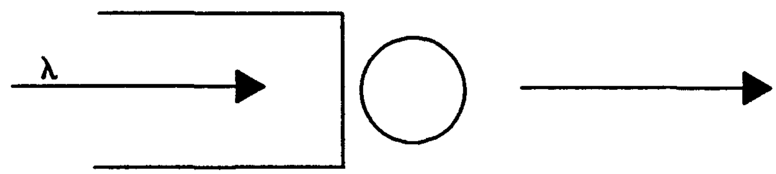
Figure 3. Network of queues.

SIMPLE ANALYSIS TECHNIQUES

Given here are simple algorithms for solving a basic queueing system. These algorithms illustrate MVA concepts. POD formulas are given by an Open System or a Closed System.

Open System

A Queueing Network is open if customers arrive from outside the system. Arrival rate denotes arrival intensity. See figure 4.



λ: arrival rate

Figure 4. Open system.

Throughput

$$X_c = \lambda_c.$$

Throughput equals arrival rate if utilization <= 1.

$$U_{c,d} = \lambda_c \ S_{c,d}.$$

Utilization of class c at device d equals arrival rate of class c customers times class c service demand at device d.

Response Time

$$R_{c,d} = \frac{S_{c,d}}{1 - \sum_{c} U_{c,d}} .$$

Response time of class c at device d equals service demand of class c at d over 1 minus the sum of the utilizations of each class c at d (as $0 \leq \Sigma\, U_{c,d} \leq 1$, $1 - \Sigma\, U_{c,d}$ is unused capacity of d).

Therefore, class c system response time

$$R_c = \sum_{d=1}^{D} R_{c,d}$$

is the sum of class c response times over all devices d, where D is the total number of devices in the system.

Closed System

A Queueing Network is closed if customers are generated inside the system (such as in a terminal system). In closed classes, the number of class c terminals (N_c) and think time for each class c terminal denote workload intensity (Z_c). Figure 5 is an example of a closed system with N terminals and one queueing center.

Solution Techniques

Response Time

Given $I_{c,d}(N) = Q_{c,d}(N - 1_c)$, that is, the class c Arrival Instant Queue length of device d with N total terminals is equal to the queue length of class c at d with one less class c terminal, then

$$R_{c,d}(N) = S_{c,d}(1 + I_{c,d}(N));$$

that is, the response of class c time at device d with N terminals is equal to the class c service demand at d times the quantity of 1 plus the class c arrival instant queue length of d with N terminals.

Throughput

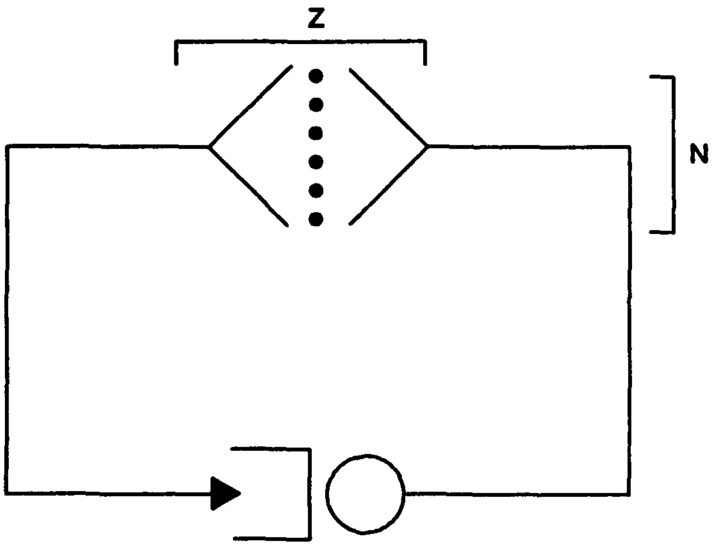
$$X_c(N) = \frac{N_c}{Z_c + \sum_{d=1}^{D} R_{c,d}(N)} .$$

Class c throughput with N terminals equals the number of class c terminals over class c think time plus the sum of class c response times.

Queue Length

$$Q_{c,d}(N) = X_c(N)\, R_{c,d}(N).$$

This equation is the queue length at device d of the class c terminals.



N: number of terminals
Z: think time

Figure 5. Closed system.

POD ALGORITHM

The main algorithm for POD is an Approximate MVA for mixed models. A mixed model contains techniques for solving both open and closed queueing networks. The POD algorithm also uses an approximation technique for determining Arrival Instant Queue length. The Approximation technique is required because Exact techniques have large memory and computation demands.

Approximation Technique for Arrival Instant Queue Length of Closed Systems

The basic MVA algorithm for response time R is

$$R_{c,k} = S_{c,k}\ (1 + Q_{c,k}).$$

Queue length Q is required but is hard to obtain exactly. Therefore, an Approximation technique is needed. The Queue Length Approximation technique is useful because it allows analysis of a queueing network (a representation of a computer system) using MVA techniques without the large memory and computational demands associated with Exact MVA techniques. The Approximation, together with MVA (known as Approximation MVA), estimate the queue length at a device (for all classes served at that device). This estimate is used as a parameter for the response time equation.

The algorithm is

given: N_c = number of class c terminals
D = number of devices

1. For all c, d

   c = (0 ... 0) with c zero elements
   d = (0 ... 0) with d zero elements

$$Q_{c,d} = \frac{N_c}{D}.$$

   Initialize $Q_{c,d}$ to the number of users divided by the number of devices.

2. $$A_{c,d}(N) = \sum_{j=1}^{c} Q_{j,d}(N - 1_c) \quad \text{where} \quad \sum_{j=1}^{c} Q_{j,d}(N) = \frac{1}{N_c}{}'Q_{c,d}(N).$$

   Approximate Arrival Instant Queue length by estimating the Queue length with minus one class c terminal. This should give us a view of the queue that an arriving customer will observe.

POD MIXED MODEL ALGORITHM

A simpler version of this algorithm is found in *Quantitative System Performance* by Edward D. Lazowska et al. (1984), that evaluates a mixed system model. Using this algorithm, POD is able to analyze systems that have both open classes (external input) and closed classes (such as cycle events of keyboard input). The basic approach is to determine the utilization caused by the open classes (this can be immediately obtained by O, open class) and "hiding" that utilization from the closed classes. This is done through a technique known as "inflation" and (as a mathematical convenience) increases the service time of the closed class proportional to the amount of utilization "hidden" from them. The algorithm follows.

Let O be the set of open classes and C be the set of closed classes.

1. Obtain the utilization for each device in the open classes:

$$U_{c,d} = \lambda_c' D_{c,k} \quad , \quad c \in \{O\}.$$

Total utilization is then

$$U\{O\} = U_C = \sum_{c}^{C} \lambda_c S_{c,d} \quad ,$$

2. Solved closed model by

   a. Approximate Arrival Instant Queue lengths A

   b. Determine Response Time, Throughput, and Queue length.

$$R_{c,d} = S'(1 + Q_{c,d}) \quad , \text{ using } S' \text{ when } S'_{c,d} = \frac{S_{c,d}}{1 - U\{O\}} \; .$$

$$X_c = \frac{N_c}{Z_c + \sum_{d=1}^{d} R_{c,d}} \; .$$

   c. Queue length

$$Q_d = \sum_{c=1}^{C} X_c R_{c,d} \; .$$

   d. Repeat steps b and c until subsequent estimates for Queue length converge within 0.2%.

Resolve the open model using Queue length determined in closed model and utilization used in open model.

$$R_{c,d} = \frac{S_{c,d}\,(1 + Q_d\{C\})}{1 - U_d\{O\}} \quad , \quad C \in \{O\} \; .$$

$$Q_{c,d} = \lambda_c' R_{c,d} \quad , \quad c \in \{O\} \; .$$

APPROACH FOR MODELING SYNCHRONIZATION IN ADA-BASED SYSTEMS: RENDEZVOUS ANALYSIS

ADA-based systems have (as a language feature) primitive mechanisms for specifying synchronization among tasks. Synchronized communication of data is known as Rendezvous. Since communication is a very important component of Distributed Systems, it is important to know through analysis the delays that may occur during Rendezvous. (See figure 6.)

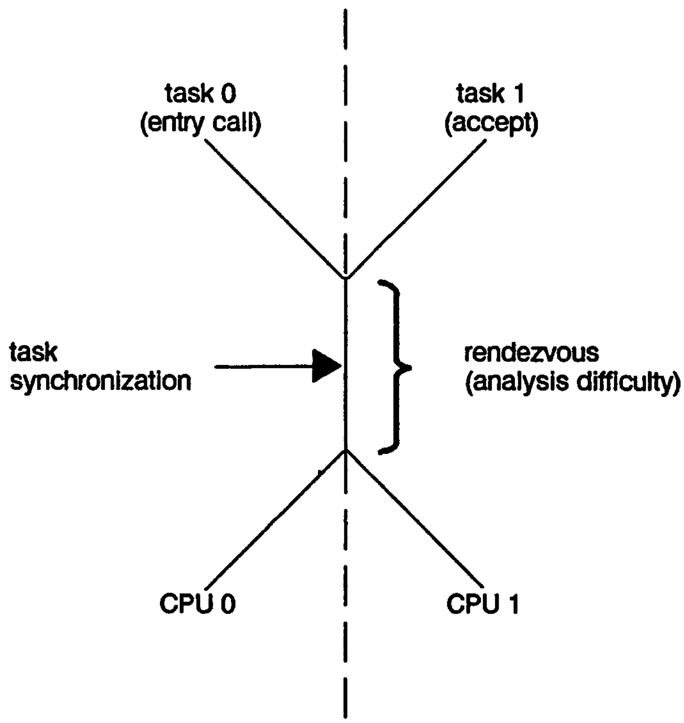


Figure 6. Modeling synchronization.

ANALYSIS OF RENDEZVOUS

The Analysis of Rendezvous is complex due to the number of variables needed to parametrize a realistic State Space Balance model. A rendezvous is a synchronous communication between two tasks. For a rendezvous to occur, the sending task initiates an Entry Call (ADA construct for requesting a rendezvous) and the receiving task ACCEPTS (another ADA construct) the rendezvous. The complexity of this problem is increased because there can be several tasks attempting a rendezvous with a single Accepting task. Since only one Entry Call can be serviced at a time, the others queue (wait) in first-come, first-serve (FCFS) order. Also, the Accepting task may have contention for its computing resource. A queueing formation of this problem can view the Accepting task as a server with the rate of service varying inversely with the total system utilization and the Entry Call as workload class arriving.(See figure 7.)

The problem with this approach is that the server (Accepting task) execution rate varies with utilization. The State Space Balance Equation solution yields a very complex three-dimensional Markov chain. Solving this Markov chain is infeasible due to space and time requirements. Developing a State Space Balance Equation does give insight to approximation techniques that could be used.

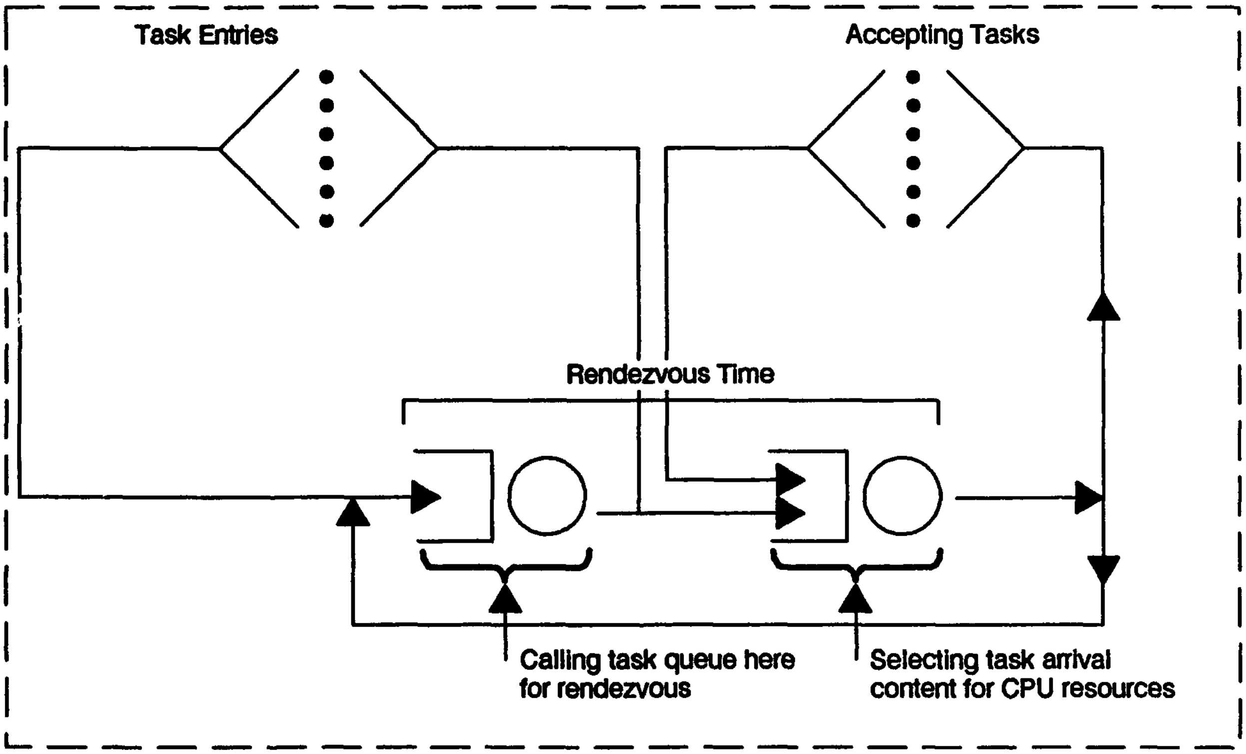


Figure 7. Queueing view of rendezvous.

DISTRIBUTED MVA (DMVA) SOLUTION

The basic Distributed Mean Value Analysis (DMVA) algorithm uses an "Active" Flow Equivalent Service Center whose service rate is inversely proportional to the utilization of its CPU. The following (figure 8) is an algorithm using DMVA for analyzing a rendezvous.

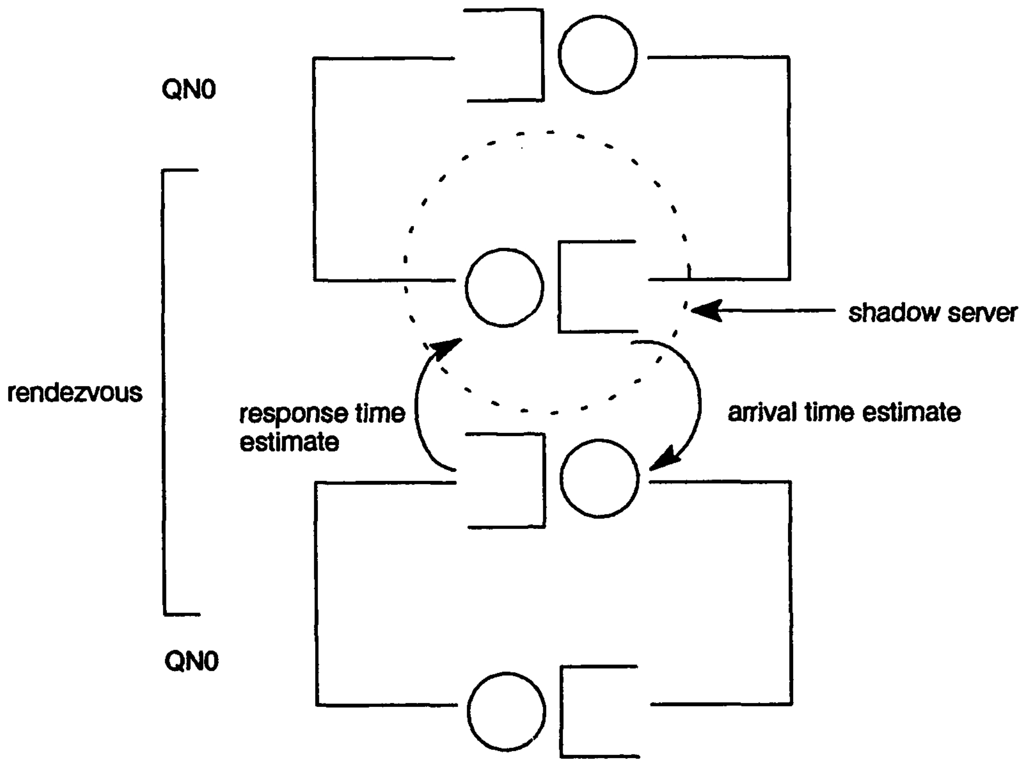


Figure 8. Distributed Mean Value Analysis (DMVA) solution.

Assumptions

The system is composed of two CPUs with the Entry Call residing on one CPU and the Accepting task (along with other jobs) on the second CPU.

Algorithm

1. Solve the Queueing Network for the Entry Call's system (QN0) by replacing the Accepting task with a dummy or shadow device.

2. Solve the Queueing Network for the Accepting task's system (QN1) (1) by using QN0 throughput as the arrival rate for the Accepting task, and (2), by using the utilization "hiding" method and subtracting the utilization caused by other classes.

3. Iterate from step 1 until QN0 and QN1 throughputs converge.

SUMMARY

This technical document describes some mathematical techniques for using the Performance Oriented Design (POD) tool. Response Time, Throughput, and Utilization techniques for computer systems are used. Mean Value Analysis (MVA) analyzes Queueing Networks that represent computer systems. Response Time, Throughput, Queue Length, and Device Utilization are used as performance measures.

The document also describes how to solve basic queueing systems by using simple algorithms such as the POD algorithm for a Mixed Model System. Modeling Synchronization in ADA-based systems is explained using the Rendezvous Analysis technique. And, lastly, an explanation of the basic Distributed Mean Value Analysis (DMVA) algorithm is given.

REFERENCES

Agrawal, S. C. 1985. *Metamodeling*, MIT Press, Cambridge, MA.

Lazowska, E., J. Zahorian, et al. 1984. *Quantitative System Performance*, Prentice-Hall, Englewood Cliffs, NJ.

Rieser, M. and S. S. Lavenberg. 1980. "Mean Value Analysis of Closed Multichain Queueing Networks," *ACM*, April 1980, pp. 113-323.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1990	Final

4. TITLE AND SUBTITLE
MATHEMATICAL TECHNIQUES FOR THE PERFORMANCE ORIENTED DESIGN (POD) TOOL

5. FUNDING NUMBERS
PE: 0602234N
WU: DN088690

6. AUTHOR(S)
A. Sterrett

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Ocean Systems Center
Code 411
San Diego, CA 92152-5000

8. PERFORMING ORGANIZATION REPORT NUMBER
NOSC TD 1936

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Naval Ocean Systems Center
Code 40
San Diego, CA 92152-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Mathematical techniques for the NOSC Performance Oriented Design (POD) tool are described using Response Time, Throughput, and Utilization techniques for computer systems. Queueing Networks that represent computer systems are analyzed using Mean Value Analysis (MVA) with performance measures such as Response Time, Throughput, Queue Length, and Device Utilization. Simple algorithms for solving basic queueing systems are included such as the POD algorithm for a Mixed Model System. Modeling Synchronization in ADA-based systems is explained using the Rendezvous Analysis technique. The document concludes with an explanation of the basic Distributed Mean Value Analysis (DMVA) algorithm.

14. SUBJECT TERMS
Mean Value Analysis (MVA)
queue length
approximation technique
Distributed MVA (DMVA)
mixed model algorithm
inflation
rendezvous analysis

15. NUMBER OF PAGES
19

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAME AS REPORT

INITIAL DISTRIBUTION

Code 0012	Patent Counsel	(1)
Code 0142	K. Campbell	(1)
Code 0144	R. November	(1)
Code 402	R. A. Wasilausky	(1)
Code 41	A. Justice	(1)
Code 411	D. Hayward	(1)
Code 411	L. Sutton	(1)
Code 411	R. Holmes	(1)
Code 411	A. Sterrett	(4)
Code 921	J. Puleo	(1)
Code 961	Archive/Stock	(6)
Code 964B	Library	(3)

Defense Technical Information Center
Alexandria, VA 22304-6145 (4)

NOSC Liaison Office
Washington, DC 20363-5100 (1)

Center for Naval Analyses
Alexandria, VA 22302-0268 (1)

Office of Naval Technology
Arlington, VA 22217-5000 (1)

Naval Weapons Center
China Lake, CA 93555-6001 (1)

Naval Underwater Systems Center
Newport, RI 02841 (1)

Naval Research Laboratory
Washington, DC 20375-5000 (1)

Naval Surface Warfare Center
Silver Spring, MD 20390-5000 (1)

Defense Advanced Research Projects Agency
Arlington, VA 22209-2308 (2)

Office of Naval Research
Arlington, VA 22217-5000 (3)

Space & Naval Warfare Systems Command
Washington, DC 20363-5100 (1)

Naval Air Development Center
Warminster, PA 18974-5000 (2)

Naval Postgraduate School
Monterey, CA 93943 (1)

Carnegie-Mellon University
Pittsburgh, PA 15213 (1)